

Amendments to the Claims

1. (original) A method of constructing at least one computer program that solves a problem, comprising the steps of:
 - defining a set of traits in which each trait characterizes a portion of a solution algorithm to the problem;
 - defining a programming interface for at least one of the traits;
 - providing an implementation for at least one of the defined programming interfaces;
 - specifying a subtrait associated with at least one of the traits or the implementations;
 - selecting a top-level trait that characterizes a solution to the problem;
 - selecting a top-level implementation for the top-level trait;
 - selecting an implementation for each subtrait required for the top-level trait or the top-level implementation;
 - recursively selecting an implementation for each subtrait associated with at least one of the traits or the implementations in order to construct a trait hierarchy that forms a computer program for solving the problem.

2. (original) The computer program constructing method according to claim 1, wherein the at least one trait comprises a plurality of traits.

3. (original) The computer program constructing method according to claim 2, wherein a computer programming interface is defined for each of the traits.

4. (original) The computer program constructing method according to claim 3, wherein an implementation is provided for each computer programming interface.
5. (original) The computer program constructing method according to claim 1, wherein the subtrait comprises a plurality of subtraits.
6. (original) The computer program constructing method according to claim 4, wherein the subtrait comprises a plurality of subtraits.
7. (original) The computer program constructing method according to claim 1, wherein the top-level trait comprises a plurality of top-level traits.
8. (original) The computer program constructing method according to claim 6, wherein the top-level trait comprises a plurality of top-level traits.
9. (canceled)
10. (original) The computer program constructing method according to claim 5, wherein the subtraits are associated with at least one of the traits, the implementation, or both.
11. (original) The computer program constructing method according to claim 1,

wherein the subtrait is one of the defined set of traits.

12. (original) The computer program constructing method according to claim 6,

wherein the subtraits associated with the traits, the implementation, or both.

13. (original) The computer program constructing method according to claim 4,

further comprising the steps of:

implementing an evaluation module that executes a constructed computer program in order to determine its effectiveness in solving the problem; and

applying an optimization technique that carries out the steps of claim 1 to generate at least one computer program that solves the problem, and that uses feedback from the evaluation module to generate at least one additional computer program that better solves the problem.

14. (original) The computer program constructing method according to claim 13,

wherein the optimization technique is selected from the group consisting of simulated annealing, an evolutionary algorithm, and a particle swarm optimization.

15. (original) The computer program constructing method according to claim 13,

further comprising the steps of: allowing a user to interactively choose which trait implementations are favored or excluded at each point in each alternative computer program created by the optimization technique.

16. (original) The computer program constructing method according to claim 1, further comprising the steps of:

defining at least one self-describing method as part of the trait implementation's interface that provides information about the trait implementation or its associated subtrait; and

implementing the at least one self-describing method as part of the trait implementation.

17. (original) The computer program constructing method according to claim 16, further comprising the steps of:

using the at least one self-describing method in a user interface to provide descriptions and other detailed information about the constructed solution algorithm.

18. (original) The computer program constructing method according to claim 16, further comprising the steps of:

using the at least one self-describing method in an optimization technique to assist in the creation of alternative computer programs.

19. (original) The computer program constructing method according to claim 16, further comprising the steps of:

using the at least one self-describing method in an interactive development environment to assist a user in assembling computer programs.

20. (original) A system for constructing at least one computer program that solves a problem, comprising:

means for defining a set of traits in which each trait characterizes a portion of a solution algorithm to the problem;

means for defining a programming interface for at least one of the traits;

an implementation for at least one of the defined programming interfaces;

a subtrait associated with at least one of the traits or the implementations;

means for selecting a top-level trait that characterizes a solution to the problem;

means for selecting a top-level implementation for the top-level trait;

means for selecting an implementation for each subtrait required for the top-level trait or the top-level implementation;

means for recursively selecting an implementation for each subtrait associated with at least one of the traits or the implementations in order to construct a trait hierarchy that forms a computer program for solving the problem.